ISSN 1749-3889 (print), 1749-3897 (online)
International Journal of Nonlinear Science
Vol.6(2008) No.1,pp.86-96

A Generalized Method of Wavelet Importance Sampling for Dynamic Global Rendering

Zhongwei Chen[1], Qin Wang[1], Lifeng Xi[2] *
[1] College of Information Technology, Zhejiang Wanli University, Ningbo, 315100, P. R. China
[2] Institute of Mathematics, Zhejiang Wanli University, Ningbo, 315100, P. R. China
(Received 4 November 2007, accepted 18 March 2008)

Abstract: An optimal wavelet product representation is employed to reduce computation by a strategy for hierarchically sampling a wavelet tree. For dynamic global rendering with multiple occlusions, we extend this double triple product integral to generalized multi-function product integral by proper ways of factorization that is reasonable for one dynamic variable applications of rendering. Final experiments are well implemented with a GPU enabled pipeline with dynamic objects.

Keywords: Monte Carlo techniques; global illumination; wavelet product integral

1 Introduction

Computing the illumination in a virtual scene is one of the basic problems of CG. On the other hand, algorithms which are capable of capturing the subtle gradations due to indirect illumination and area light sources require considerably larger resources[Fan X.,Phong B.T. etc.][11][17][20]. High fidelity images based on a whole range of reflection phenomena described by the rendering equation

$$L_o(x,\vec{\omega}_o) = L_e(x,\vec{\omega}_o) + \int\limits_{\Omega} f_r(x,\vec{\omega}_i,\vec{\omega}_o) L_i(x,\vec{\omega}_i)\cos(\theta_i) d\vec{\omega}_i \quad (1)$$

often take hours or days to compute[J. T. Kajiya 1986][3].

The high complexity of such computations is a consequence of the fact that each surface in a scene can potentially affect the appearance of every other surface, leading to very large and dense linear systems [Tian L. etc.] [8].

The performance can be improved if we incorporate knowledge about the function being integrated into the sampling process. The idea is to concentrate samples to parts of the function where it is likely to be large. This technique is called *importance sampling*, and can vastly reduce the variance in Monte Carlo techniques[Malvin H. Kalos etc.][4].

Several researchers have recently worked on this problem, by either combining samples drawn independently according to the importance of the illumination and the BRDF [Agrawal, S.][7] [Andrew S. Glassner][10], or more recently, by drawing samples from the product distribution of the illumination and the BRDF [Burke D. etc.][1] [2] [Clarberg, P.,Xi L. etc] [5] [6]. These approaches produce high quality images with a small number of samples in unoccluded regions.

Recently, Clarberg et al. presented an algorithm called Wavelet Importance Sampling (WaIS) that samples products of wavelet functions[5]. Their algorithm uses a property of wavelets that allows a wavelet product to be evaluated in a top-down fashion, and they introduced a warping technique that transforms a uniform distribution of points to the product distribution using the wavelet product as a guide. However, WaIS addressed only aspects of direct illumination and static ones.

*Corresponding author. Tel. : +86-574-8822 2249; fax: +86-574-8822 2090. *E-mail address*: xilf@zwu.edu.cn.

In this paper, we present an improvement method of WaIS for real-time rendering with dynamic objects under global illumination.

2 Product importance sampling for rendering

2.1 BRDF

The bidirectional reflectance distribution function (BRDF) characterizes the reflection of light on a surface. The BRDF was first formally defined by Nicodemus et al. [9]. In radiometric terms, the BRDF is the surface radiance divided by the surface irradiance, i.e., the incident light flux per unit illuminated surface area:

$$f_r(x,\vec{\omega}_i,\vec{\omega}_o)=\frac{dL_o(x,\vec{\omega}_o)}{dL_i(x,\vec{\omega}_i)}=\frac{dL_o(x,\vec{\omega}_o)}{L_i(x,\vec{\omega}_i)\cos(\theta_i)d\vec{\omega}_i}, \tag{2}$$

where x is the surface position, $\vec{\omega}_i$ is the incident direction, $\vec{\omega}_o$ is the outgoing (viewing) direction, $L_i(x,\vec{\omega}_i)$ is outgoing radiance, and $L_o(x,\vec{\omega}_o)$ is incident radiance (irradiance). See Figure 1 for an illustration.

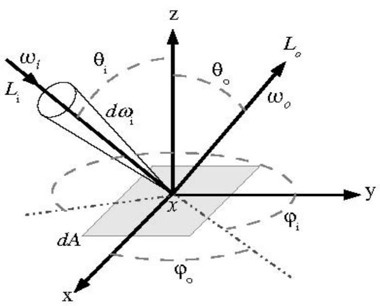

Figure 1: The bidirectional reflectance distribution function (BRDF) describes the ratio of outgoing radiance to incident radiance (irradiance).

Theoretically, the BRDF also depends on other variables such as wavelength and polarization of the light, but we are usually considering only unpolarized light of one specific wavelength at the time. Therefore, the BRDF can be written as a four-dimensional function. The function is typically parameterized over the spherical coordinates for the incident direction $\vec{\omega}_i=(\theta_i,\varphi_i)$ and outgoing direction $\vec{\omega}_o=(\theta_o,\varphi_o)$.For a detailed discussion of the BRDF and its use in computer graphics, see the books by Glassner [10].

2.2 Direct illumination

The global illumination rendering equation (1) can be split into several components:

$$L_o(x,\vec{\omega}_o)=L_{dir}(x,\vec{\omega}_o)+L_{ind}(x,\vec{\omega}_o)+L_e(x,\vec{\omega}_o), \tag{3}$$

where L_{dir} is the direct illumination, L_{ind} is the indirect illumination, and L_e is the self-emitted radiance. Here, the direct illumination is given by the integral:

$$L_{dir}(x,\vec{\omega}_o)=\int_{\Omega} f_r(x,\vec{\omega}_i,\vec{\omega}_o)L(x,\vec{\omega}_i)v(x,\vec{\omega}_i)\cos(\theta_i)d\vec{\omega}_i, \tag{4}$$

where the incident radiance, $L(x,\vec{\omega}_i)$, is provided by light sources in the scene, and $v(x,\vec{\omega}_i)$ is the visibility of a light source in direction $\vec{\omega}_i$. In order to apply realistic lighting to a virtual scene, it is common to capture real lighting in a high-dynamic range environment map [Matt Pharr,etc.] [12], and use that for L during rendering.

2.3 Product importance sampling estimator

A common approach to evaluate the direct lighting equation (4) is to use Monte Carlo integration, which replaces the continuous integral with the average of N Monte Carlo samples [3].

Many importance sampling techniques for direct lighting concentrate on either sampling the light source or the BRDF. BRDF importance sampling is better suited for specular materials, while environment map importance sampling is better for diffuser BRDFs.

To address this problem, Veach and Guibas [14] presented a novel technique for combining estimators in Monte Carlo methods using multiple importance sampling, which is a powerful method for the case where either the lighting or the BRDF is complex, as it will pick the best of the available sampling techniques. However, when both the lighting and the BRDF are complicated, their technique provides a smaller advantage. Burke et al. [1] introduced a technique for rendering objects with complex materials illuminated by an environment map.

In their work, the aim is to perform importance sampling using the *product* of the incident light distribution and the BRDF as the importance function:

$$p(\vec{\omega}_i) = \frac{f_r(\vec{\omega}_i, \vec{\omega}_o) L_i(\vec{\omega}_i)\cos(\theta_i)}{\int\limits_{\Omega} f_r(\vec{\omega}_i, \vec{\omega}_o) L_i(\vec{\omega}_i)\cos(\theta_i) d\vec{\omega}_i}. \tag{5}$$

Observe that the normalization term in the denominator is the direct illumination integral with the visibility term $v(\vec{\omega}_i)$ omitted. In other words, this term is the exitant radiance in the absence of shadows. Burke et al. refer to it as *Lns :* radiance, no shadows. [1]:

$$L_{ns} := \int\limits_{\Omega} f_r(\vec{\omega}_i, \vec{\omega}_o) L_i(\vec{\omega}_i)\cos(\theta_i) d\vec{\omega}_i. \tag{6}$$

If sample directions $\vec{\omega}_i^{(j)}$ are drawn according to the product distribution in Equation 6, then Equation 7 can be used to estimate $L_{N,p}(\vec{\omega}_o)$ as:

$$L_{N,p}(\vec{\omega}_o) = \frac{1}{N}\sum_{j=1}^{N} \frac{f_r(\vec{\omega}_i^{(j)}, \vec{\omega}_o^{(j)}) L_i(\vec{\omega}_i^{(j)}) v(\vec{\omega}_i^{(j)})\cos(\theta_i)}{p(\vec{\omega}_i^{(j)})} = \frac{L_{ns}}{N}\sum_{j=1}^{N} v(\vec{\omega}_i^{(j)}). \tag{7}$$

$L_{N,p}(\vec{\omega}_o)$ is referred to as the *bidirectional estimator* for the direct illumination integral.

Note that the variance of the bidirectional estimator for the reflected radiance is proportional to the variance in the visibility function, $\vec{\omega}_i^{(j)} \propto p(\vec{\omega}_i), j = 1, ..., N$. This is an improvement over sampling techniques that only consider either the illumination or the BRDF in the sampling process. This is because the variance of these techniques depends in addition on the variance in the function that they do not sample from, BRDF or illumination respectively.

3 Product importance sampling using wavelet

We focus on the efficient computation of the multi-function product integral and the product of multiple functions now. For this purpose, we choose Haar bases as the basis set $\boldsymbol{B}$. Compared with the pixel domain representation, wavelets allow us to approximate signals at low distortion with a small number of significant coefficients. Haar bases, amazingly, have an interesting property that simplifies the computation as many of the integral coefficients are zero [S. G. Mallat] [23]. Therefore, by using only non-zero wavelet coefficients and non-zero integral coefficients, evaluation of both the multi-function product integral and the product of multiple functions can be significantly accelerated.

3.1 2D Haar Bases

Nonstandard Haar wavelet transform [Stollnitz et al. 1996] [15] decomposes a $2^n \times 2^n$ image into a 2D signal with $2^n \times 2^n$ coefficients. Each coefficient corresponds to a basis function defined in the region

$< j, t_1, t_2 >$ where j is the *scale* ($0 \leq j < n$), t_1 and t_2 are spatial translations ($0 \leq t_1, t_2 < 2^j$). In each region $< j, t_1, t_2 >$, four normalized 2D Haar basis functions are defined:

(1) $\phi^j_{t_1,t_2}$ *normalized Haar scaling basis function:* $\phi^j_{t_1,t_2}(x, y) = 2^j \phi^0(2^j x - t_1, 2^j y - t_2)$ where ϕ^0 is the *mother scaling function*.

(2) $\psi^j_{t_1,t_2}$ *normalized Haar wavelet basis function.* There are three types of wavelets defined in the region $< j, t_1, t_2 >$:

$$\psi^j_{m,t_1,t_2}(x, y) = 2^j \psi^0_m(2^j x - t_1, 2^j y - t_2), m = 1, 2, 3$$

where $\psi^0_m, m = 1, 2, 3$, are three different mother wavelets, denoting the horizontal, vertical and diagonal differences.

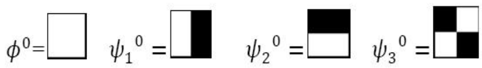

Figure 2: The mother scaling function ϕ^0 and the three mother wavelet functions $\psi^0_m, m = 1, 2, 3$. The functions are +1 where white and -1 where black, and implicitly zero outside the unit square.

If set vector $\mathbf{t} = (t_1, t_2)$, and consider $F(x, y)$ as a two-dimensional image with $2^l \times 2^l$ pixels, then two-dimensional image $F(x, y)$ can be further expressed as a sum of the first scaling function plus the wavelet functions as:

$$F = F^0_{0,0}\phi_{0,0} + \sum_{k=0}^{l-1} \sum_t \sum_m F^m_{k,t} \psi^m_{k,t} = \sum_i F_i \psi_i. \tag{8}$$

3.2 General 2D Wavelet Product

Given two functions expressed in an orthonormal basis, it is possible to multiply them together and get the product expanded in the same basis Ng et al. [16]

Let $G = \sum G_j \psi_j$ and $H = \sum H_k \psi_k$ be the two images represented in the Haar basis. The wavelet product, $F = \sum F_i \psi_i$, of G and H is then given by:

$$F = G \cdot H \Leftrightarrow \sum F_i \psi_i = \sum G_j \psi_j \cdot \sum H_k \psi_k. \tag{9}$$

By integrating against the i^{th} basis function, we can directly obtain the i^{th} coefficient for the wavelet representation of the product F_i as follows:

$$\begin{aligned} F_i &= \iint F(x)\psi_i(x)dx = \iint \psi_i(x)G(x)H(x)dx \qquad (10) \\ &= \iint \psi_i(x)(\sum G_j \psi_j(x) \cdot \sum H_k \psi_k(x))dx \\ &= \sum_j \sum_k G_j H_k \iint \psi_i(x)\psi_j(x)\psi_k(x)dx \\ &= \sum_j \sum_k C_{ijk} G_j H_k, \end{aligned}$$

where

$$C_{ijk} = \iint \psi_i(x)\psi_j(x)\psi_k(x)dx. \tag{11}$$

Note that these equations are valid for any domain and suitable orthonormal basis, only the tripling coefficients will differ. Due to the compact support of the Haar basis functions, most of the tripling coefficients will be zero. The non-zero coefficients are given by the Haar tripling coefficient theorem [Ng,R.,etc.][16].

Generalized Haar Integral Coefficient Theorem *The n^{th}-order Haar integral coefficient C^n has a non-zero value, if and only if the numbers of the three kinds of wavelets ψ_1, ψ_2* and ψ_3 *at the finest scale have the same parity. In this case, the integral coefficient is $\pm 2^{\Sigma j - 2j_0}$, where $\sum j$ is the sum of the scales of* *all operand basis functions, and j_0 is the scale of the finest basis function. The sign of the integral coefficient is the multiplication of the signs of the subregions of all parent basis functions that the child basis function falls into.*

Thus, the integral of three 2D Haar basis functions *Fi* is non-zero if and only if one of the following three cases holds:

1. All three are the scaling function. In this case, C_{ijk} = 1.

2. All three functions occupy the same wavelet square, and all are of different wavelet types. $C_{ijk} = 2^l$, where the square is at level l.

3. Two are identical wavelets, and the third is either the scaling function or a wavelet that overlaps at a strictly coarser level. C_{ijk} = $\pm 2^l$, where the third function exists at level l.

The tripling coefficient theorem is written in general terms, and describes the cases where the tripling coefficients are non-zero. In this application, where we are looking at a specific basis function, ψ_i, the theorem can be rewritten to make the different cases more clear:

1. ψ_i is the mother scaling function:

(a) ψ_j and ψ_k are also the mother scaling function. $C_{ijk} = 1$.

(b) ψ_j and ψ_k are identical wavelets (at any level). $C_{ijk} = 1$.

2. ψ_i is a wavelet function at level l:

(a) All three functions occupy the same wavelet square and all are of different wavelet types. $C_{ijk} = 2^l$.

(b) $\psi_j$and ψ_k are identical wavelets under the support of ψ_i and exist at a strictly finer level. $C_{ijk} = \pm 2^l$.

(c) One of the wavelets is identical to ψ_i, and the other is either the mother scaling function or a wavelet that overlaps at a strictly coarser level. $C_{ijk} = \pm 2^{l'}$, where the coarser function exists at level l'.

3.3 Wavelet importance sampling

In wavelet importance sampling, we approximate the function $f(x)$ with an image $F(x)$, expressed in the Haar wavelet basis. For simplicity, the image is defined to cover the unit square. Consider a wavelet square $s = (l, \mathbf{t})$ at level $l$and translation **t**. The square has an area of $A(s) = 2^{-l} \times 2^{-l} = 2^{-2l}$. The average function value $F(s)$ over the square, can be found by integrating the function over s. However, due to the constant and disjoint scaling functions, the average function value is given by the scaling coefficient for the square as follows:

$$F(s) = \iint\limits_{s} F(x)dx = 2^l \iint \phi^0_{l,t}(x)F(x)dx = 2^l F^0_{l,t}, \tag{12}$$

$$I = \iint F(x)dx = F^0_{0,0}. \tag{13}$$

Thus, the probability density of the square s, is given by:

$$p(s) = \frac{F(s)}{I} = 2^l \frac{F^0_{l,t}}{F^0_{0,0}}, \tag{14}$$

the probability of placing a sample at a coordinate x within the square s, should be equal to:

$$p(x \in s) = p(s)A(s) = 2^{-2l}\frac{F(s)}{I} = 2^{-l}\frac{F^0_{l,t}}{F^0_{0,0}}. \tag{15}$$

For recursive algorithms, it is useful to know the conditional probabilities for each child square, given that the parent square is sampled. Let s be the parent square at level l, and let $s_i, i = 1...4$, be the four child squares at level $l + 1$. The conditional probability for each of the four children can be expressed in the function values for the parent and child squares as:

$$p(x \in s_i \,|x \in s\,) = \frac{p(x \in s_i)}{p(x \in s)} = \frac{2^{-2(l+1)}F(s_i)/I}{2^{-2l}F(s)/I} = \frac{1}{4}\frac{F(s_i)}{F(s)}, \tag{16}$$

and similarly expressed in scaling coefficients as:

$$p(x \in s_i \,|x \in s\,) = \frac{p(x \in s_i)}{p(x \in s)} = \frac{2^{-(l+1)}F^0_{l+1,t}/F^0_{0,0}}{2^{-l}F^0_{l,t}/F^0_{0,0}} = \frac{1}{2}\frac{F^0_{l+1,t}}{F^0_{l,t}}. \tag{17}$$

3.4 Improved Sampling of Wavelet Products

For a simple case, the importance function f(x) is a product of only two functions, f(x) =g(x)h(x). We store approximations of g(x) and h(x) as images, G and H respectively, expressed as Haar wavelets. Then coefficients for the product F = G·H of the two wavelets can be computed using theory in 3.2.

In practice, as stated in last section, it is unnecessary to compute detail coefficients for the wavelet product, as only the scaling coefficients at each level are needed for sampling. So the general product in 3.2 could be simplified by direct product of only scaling coefficients. While replacing ψ_i with the specific scaling function $\phi_{l,t}$, the scaling coefficient for the product is then given by:

$$\begin{aligned} F^0_{l,t} &= \iint F(x)\phi_{l,t}(x)dx = \iint \phi_{l,t}(x)G(x)H(x)dx \\ &= \iint \phi_{l,t}(x)(\sum G_j\psi_j(x)\cdot \sum H_k\psi_k(x))dx \\ &= \sum_j\sum_k G_jH_k \iint \phi_{l,t}(x)\psi_j(x)\psi_k(x)dx \\ &= \sum_j\sum_k C'_{ijk}G_jH_k, \end{aligned} \tag{18}$$

where C'_{ijk} are modified tripling coefficients, defined as:

$$C'_{ijk} = \iint \phi_{l,t}(x)\psi_j(x)\psi_k(x)dx. \tag{19}$$

It turns out that the C'_{ijk} for a scaling function at level l are non-zero if and only if one of the following two cases holds:

1. ψ_j and ψ_k are either the mother scaling function or wavelets at strictly coarser levels, l_j and l_k.

$$C'_{ijk} = \pm 2^{l_j+l_k-l}.$$

2. ψ_j and $\psi_k$are identical wavelets under the support of $\phi_{l,t}$, and exist at the same or finer levels. $C'_{ijk} = 2^l$.

The first case corresponds to a multiplication of the scaling coefficients for G and H at level l that overlap $\phi_{l,t}$, scaled by 2^l, i.e., a multiplication of the scaling coefficients $G^0_{l,t}$ and $H^0_{l,t}$. Hence, scaling coefficients for the product as:

$$F^0_{l,t} = 2^l G^0_{l,t}H^0_{l,t} + 2^l \sum_{l'\geq l,t'\in t,m} F^m_{l',t'}H^m_{l',t'}, \tag{20}$$

where the summation is over all wavelet coefficients that are under the support of $\phi_{l,t}$. the scaling coefficients $G^0_{l,t}$ and $H^0_{l,t}$ can easily be computed separately for the two functions, using standard wavelet reconstruction from their respective wavelet coefficients.

This simplified way is much more efficient than the general one.

Once the product F can be computed, the importance sampling probability computing is as same as above equations for single function case described in 3.3.

4 Generalized method in dynamic global rending

For the real-time rendering of dynamic scenes while taking into account soft shadows and light inter-reflections, the efficient wavelet product importance sampling theory could be generalized by a factoring way [Mccool, M. D.,etc.] [18] [Ben-Artzi, A.,etc.] [19] [Ahmed,etc.] [20][Lawrence J.,etc.][22].

4.1 Multi-function product integral

Given n distinct objects in a dynamic scene, the exitant radiance B at a surface point x along view direction θ due to distant environment lighting L is the product integral over all incident directions sampled at a surrounding cubemap Ω [21]:

$$\begin{aligned} B(x,\vec{\omega}_o) &= \int_{\Omega} L(\vec{\omega}_i)O_1(x,\vec{\omega}_i)\prod_{j=2}^{n} O_j(x,\vec{\omega}_i)f_r(x,\vec{\omega}_i,\vec{\omega}_o)(\vec{N}\cdot\vec{\omega}_i)d\vec{\omega}_i \\ &= \int_{\Omega} L(\vec{\omega}_i)\tilde{O}_1(x,\vec{\omega}_i)\prod_{j=2}^{n} \tilde{O}_j(x,\vec{\omega}_i)f_r(x,\vec{\omega}_i,\vec{\omega}_o)d\vec{\omega}_i, \end{aligned} \tag{21}$$

where $\vec{\omega}_i$ is the incident direction, $\vec{N}$ is the normal at x, f_r is the BRDF, O_1 is the *local visibility* at x due to self-occlusion. $O_i(2 \le j \le n)$ is the *dynamic occlusion* at x occluded by the j^{th} neighboring object in the scene. In order to eliminate the dependance of the BRDF on the normal, the cosine term $(\vec{N}\cdot\vec{\omega}_i)$ is combined with the self visibility O_1 as $\tilde{O}_1$ as:

$$\tilde{O}_1(x,\vec{\omega}_i) = O_1(x,\vec{\omega}_i)(\vec{N}\cdot\vec{\omega}_i). \tag{22}$$

For a fixed vertex x and view direction $\vec{\omega}_o$, equation (21) can be simplified as:

$$B = \int L(\vec{\omega}_i)\tilde{O}_1(x,\vec{\omega}_i)\prod_{j=2}^{n} \tilde{O}_j(x,\vec{\omega}_i)f_r(x,\vec{\omega}_i)d\vec{\omega}_i. \tag{23}$$

It is exactly the product integral of $(n+2)$ functions:

$$B = \int \prod_{j=1}^{n+2} F_j(\vec{\omega}_i)d\vec{\omega}_i. \tag{24}$$

4.2 Factoring for dynamic radiance transfer

For dynamic radiance transferring, an effective approach to accelerating the evaluation of equation 24 is stated as follows:

$$B = \int \left[\prod_{j=1}^{n+1} F_j(\vec{\omega}_i)\right] F_{n+2}(\vec{\omega}_i)d\vec{\omega}_i = \langle T, F_{n+2}(\vec{\omega}_i)\rangle\,, \tag{25}$$

where the radiance transfer vector $T$is the product of n+1 functions as:

$$T = \prod_{j=1}^{n+1} F_j(\vec{\omega}_i). \tag{26}$$

If F_1, F_2, $\cdots$, F_{n+1} are fixed, in other words, only F_{n+2} varies (i.e., dynamic instead of static), radiance transfer vector T needs to be computed only once. Therefore, shading integral reduces to a simple double function product integral of T and F_{n+2}, which can be approximated by the wavelet importance sampling method. Here we assume that only one function in the shading integral varies. This assumption is reasonable for lighting design systems, where normally the designer adjusts only one variable at a time, and real-time feedback is highly appreciated. For example, the designer may experiment with different lighting effects by fixing view conditions and objects. The designer may also render the scene from different view conditions by fixing the lighting and the objects. Another popular operation is to fix the lighting and view conditions, and relocate a single object in the scene. As long as there is only one (note that it can be any one) varying parameter, this approach can be used to generate all-frequency shadows in real-time.

In equation (25), the product of $n+2$ functions is factored into the product of two sets, one with $n+1$ functions, and the other with only one function. More generally, this factorization has the following form:

$$B = \int \left[\prod_{j=1}^{k} F_j(\vec{\omega}_i)\right]\cdot\left[\prod_{j=k+1}^{n+2} F_j(\vec{\omega}_i)\right] d\vec{\omega}_i = \langle T_1, T_2\rangle\,, \tag{27}$$

where $T_1 = \prod_{j=1}^{k} F_j(\vec{\omega}_i)$ and $T_2 = \prod_{j=k+1}^{n+2} F_j(\vec{\omega}_i)$. As a result, the product of $n+2$ functions reduces to the double function product integral of two radiance transfer vectors.

5 Experiment results

5.1 Parameterisation and wavelet transform of BRDF

By a change of variables, the BRDF can be transformed into a function that is more compact. There are many options for such reparameterizations. In our application, we need a parameterization that is suitable for both the BRDF and for the environment map. In our application, the BRDF is centered about the reflection vector $\vec{\omega}_r = (\theta_r, \varphi_r)$, instead of around the surface normal $\vec{N}$.

A non-standard approach of wavelet transform for f_r is employed here (Figure 3).

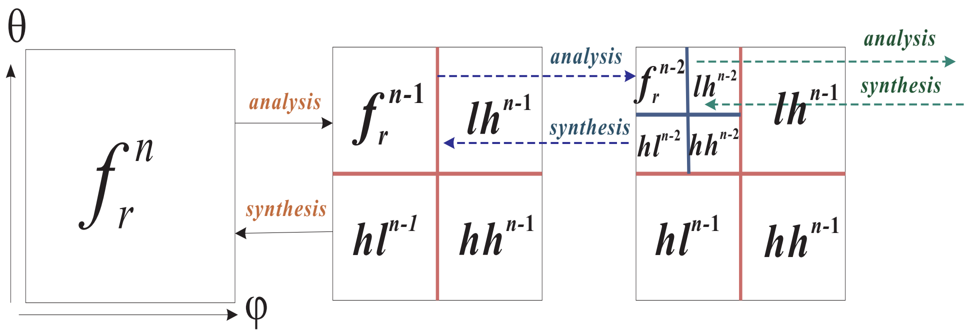

Figure 3: 2D Wavelet transform applied on each hemisphere of the original BRDF data.

5.2 Localization of EM coordinates

In our application, the BRDF is given in local coordinates with respect to the reflection vector, while an environment map is commonly expressed in global coordinates.

By rewriting the environment map as a four-dimensional function $L(\vec{\omega}, \vec{\omega}_r)$, where the direction $\vec{\omega}$ is given with respect to $\vec{\omega}_r$, the environment map is in the same local space as the BRDF [Afrouzi G. A.][24]. In practice, both the BRDF and the environment map are tabulated as a sparse 2D set of 2D wavelet compressed images. The maps are stored at the resolution 64×64 or 128×128.

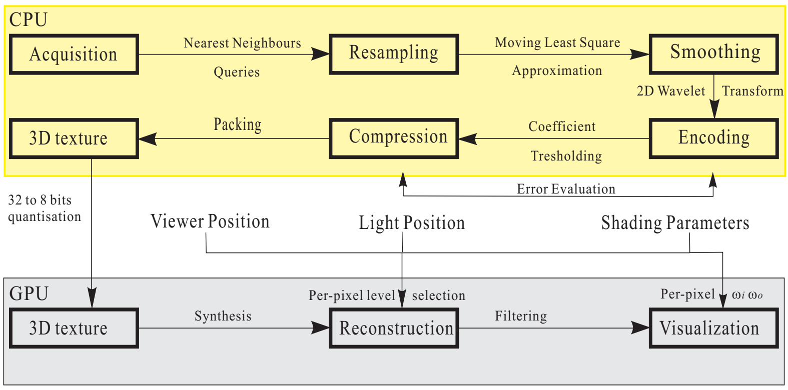

Figure 4: The GPU enhanced rendering pipeline.

5.3 Rendering pipeline with GPU shading

At the rendering time, the number of texture accesses for each fragment depends on the resolution chosen on-the-fly for the BRDF reconstruction. If the surface is far away from the viewpoint, fewer levels are required to estimate the BRDF and the performance is enhanced. These results are achieved with GPU built-in hardware filtering and by using the linearity and the multi-resolution of the wavelet encoding.

5.4 Results

A ray tracing rendering using BRDF in Cornell Box is implemented as below. The small color box make the back wall shinier with four bounces of global illumination. Notice the correct color bleeding in the scene, and correct glossy reflections of the scene in the back wall of right in figure 5. Figure 6 shows another rendering with dynamic objects of a ring on the ground environment map.

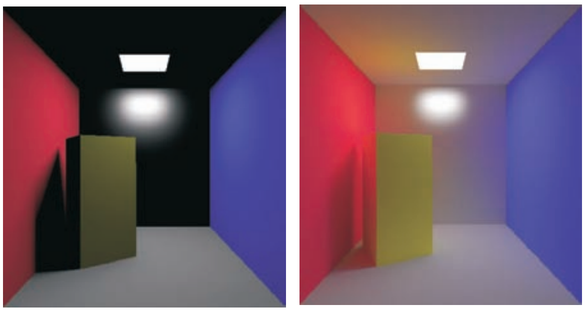

Figure 5: Left: Direct lighting; Right: Global Lighting

Figure 6: Left: direct lighting; Middle: global lighting; Right: the ground is added for rendering (dynamic)

5.5 Conclusions and future work

Wavelet product importance sampling is an efficient way for static direct illumination with complex environment mapping. According to the feature of its product sampling of two functions, the factoring scheme we developed makes shading integral reduce to a simple double function product integral. Such way is suitable for dynamic global lighting situations with multiple objects where normally only one variable is adjusted at a time, and real-time feedback is highly appreciated.

A GPU enabled pipeline is also used to accelerate the real-time rendering and worked well in practice.

Wavelet representations of BRDF and EM provide novel approaches for complex rendering. The method we proposed here can be used in other domains where the efficient computation and real-time generation are critical such as game, animation, and simulation.

Acknowledgement:

The authors thank An-Bo Le for helpful comments and suggestions.

References

[1] Burke D., Abhijeet, Heidrich W: Bidirectional Importance Sampling for Illumination from Environment Maps. In *ACM SIGGRAPH Technical Sketches*(2004)

[2] Burke D., Ghosh A., Heidrich W: Bidirectional importance sampling for direct illumination. In *Rendering Techniques 2005 Eurographics Symposium on Rendering* (Aire-la-Ville, Switzerland, 2005), Eurographics Association. 139-146.

[3] James T. Kajiya: The Rendering Equation. In *Computer Graphics (Proceedings of ACM SIGGRAPH* 86), 143–150(1986)

[4] Malvin H. Kalos ,Paula A: Whitlock. *Monte Carlo Methods, Volume 1: Basics*. Wiley-Interscience, New York(1986)

[5] Clarberg, P., Jarosz, W., Akenine-Moller, T., Jensen, H. W: Wavelet importance sampling: Efficiently evaluating products of complex functions. *ACM Transactions on Graphics (SIGGRAPH '05) 24*: 3, 1166–1175(2005)

[6] Lifeng Xi, Liangbin Zhang: A Study of Fractal Image Compression based on an Improved Genetic Algorithm. *International Journal of Nonlinear Science*. 3(2):116-124(2007)

[7] Agrawal, S., Ramamoorthi, R., Belongie, S., Jensen, H.W: Structured importance sampling of environment maps. In *SIGGRAPH 03*, 605–612(2003)

[8] Lixin Tian, Xiuming Li: Well-posedness for a New Completely Integrable Shallow Water Wave Equation. *International Journal of Nonlinear Science*. 4(2):83-91(2007)

[9] Nicodemus F. E., Richmond J. C., Hsia J. J., Ginsberg I. W., Limperis T: Geometric Considerations and Nomenclature for Reflectance, volume NBS Monograph 160. National Bureau of Standards, Washington, D.C.(1977)

[10] Andrew S. Glassner: Principles of Digital Image Synthesis. Morgan Kaufmann(1995)

[11] Xinghua Fan, Zhen Zhang, Tieniu Lu: Explicit Solitary Solutions to the Toda Lattice Equation. *International Journal of Nonlinear Science*. 3 (2): 111-115(2007)

[12] Matt, P.,Humphreys,G: Physically Based Rendering. Morgan Kaufmann(2004)

[13] Ng, R., Ramamoorthi, R., Hanrahan, P: All frequency shadows using non-linear wavelet lighting approximation. ACM TOG (SIGGRAPH 2003) 22(3):376–381(2003)

[14] Eric Veach Leonidas J. Guibas: Optimally Combining Sampling Techniques for Monte Carlo Rendering. In Proceedings of ACM SIGGRAPH 95: 419–428(1995)

[15] Eric J. Stollnitz, Tony D. DeRose, David H. Salesin: Wavelets for Computer Graphics: Theory and Applications. Morgan Kaufmann(1996)

[16] Ng, R., Ramamoorthi R., Hanrahan,P: Triple Product Wavelet Integrals for All-Frequency Relighting. ACM Transactions on Graphics, 23(3): 477–487(2004)

[17] Phong, B. T: Illumination for computer generated pictures. *Commun. ACM 18*, 6, 311–317(1975)

[18] Mccool, M. D., Ang, J., Ahmad, A: Homomorphic factorization of BRDFs for high-performance rendering. In *SIGGRAPH 01*, 185–194(2001)

[19] Ben-Artzi, A., Overbeck, R., Ramamoorthi, R: Real-time BRDF editing in complex lighting. *ACM Transactions on Graphics (SIGGRAPH 06) 25*, 3, 945–954(2006)

[20] Ahmed Hassan Ahmed Ali1, Kamal Raslan Raslan: New Solutions for Some Important Partial Differential Equations. *International Journal of Nonlinear Science*.4(2):109-117(2007)

[21] Sun, W., Mukherjee, A: Generalized wavelet product integral for rendering dynamic glossy objects. *ACM Transactions on Graphics (SIGGRAPH 2006) 25*, 3, 477–487(2006)

[22] Lawrence J., Rusinkiewicz S., Ramamoorthi R: Efficient BRDF Importance Sampling using a Factored Representation. ACM Transactions on Graphics, 23(3): 496–505(2004)

[23] Mallat, S. G:A Theory for Multiresolution Signal Decomposition: The Wavelet Representation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 11(7): 674–693(1989)

[24] Afrouzi, G. A., Khademloo, S: On Adomian Decomposition Method for Solving Reaction Diffusion Equation. *International Journal of Nonlinear Science*. 2 (1):11-15(2006)